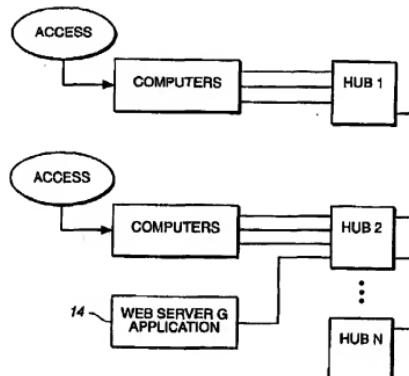




AL

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)



**FOR THE PURPOSES OF INFORMATION ONLY**

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LY	Larvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Mosam	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece	ML	Mali	TR	Turkey
BG	Bulgaria	HU	Hungary	MN	Mongolia	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MR	Mauritania	UA	Ukraine
BR	Brazil	IL	Israel	MW	Malawi	UG	Uganda
BY	Belarus	IS	Iceland	MX	Mexico	US	United States of America
CA	Canada	IT	Italy	NL	Niger	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NO	Netherlands	VN	Viet Nam
CG	Congo	KE	Kenya	NZ	Norway	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	PL	New Zealand	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	PT	Poland		
CM	Cameroon	KR	Republic of Korea	PT	Portugal		
CN	China	KZ	Kazakhstan	RO	Romania		
CU	Cuba	LK	Sri Lanka	RU	Russian Federation		
CZ	Czech Republic	LC	Saint Lucia	SD	Sudan		
DE	Germany	LI	Liechtenstein	SE	Sweden		
DK	Denmark	LK	Sri Lanka	SG	Singapore		
EE	Estonia	LR	Liberia				

## WEB-BASED NETWORK MANAGEMENT SYSTEM

Field of the Invention

5 The present invention relates generally to network management systems. More particularly, the present invention provides a method and apparatus for managing a network using a World-Wide Web server.

Background of the Invention

10 Networks frequently include large numbers of interconnected devices such as computers, and present serious difficulties to persons assigned to manage the network. Conventional network management tasks are performed by a systems administrator, who typically either must go to each device in the network or log on to an individual network device from another network device to obtain desired data about the operation of each device. Neither of these approaches are satisfactory, particularly for larger networks. Going to each 15 device in the network is a time-consuming task for a systems administrator. Logging on to an individual network device, typically performed over a telephone network link, is similarly inefficient to serve an entire network. Further, known telephone network links do not typically provide a graphical user interface to better enable the systems administrator to manage the network; and do not allow for real-time data collection and retrieval.

20 It is also known to incorporate a World-Wide Web server capability into each device in a network to enhance network management capabilities. Such an implementation is undesirable because of the expense associated with adding web server capabilities to each device, and further because many older devices cannot be modified to include web server capabilities.

25 It would be desirable for a network management system to allow a systems administrator to access multiple devices in a network to retrieve data in real time. It would further be desirable for a network management system to present a graphical user interface to a systems administrator which can be customized by the systems administrator to better manage the network. It would also be desirable for a network management system to be 30 platform-independent.

Summary of the Invention

According to exemplary embodiments of the present invention, a system for controlling a network includes one or more first control devices, such as computer terminals having a network browser capability for communicating with a common gateway interface 5 of a world-wide web server. The system further includes at least one second control device, such as a SNMP search engine associated with a web server, for communicating with each of the network devices. The common gateway interface communicates with the second control device in response to a communication received from a first control device, and the second control device retrieves information requested in the communication from selected 10 network devices and forwards the retrieved information to the requesting first control device through the common gateway interface.

The system and method of the present invention allow an administrator of a relatively complex network to manage the network remotely, obtain network data in real time from any combination of sub-networks within the network using customized HTML documents, and 15 have the retrieved data displayed in a graphical user interface. In addition, the system and method of the present invention are platform-independent.

Brief Description of the Drawings

The present invention will be more fully understood upon reading the following 20 Detailed Description of the Preferred Embodiments in conjunction with the accompanying drawings, in which like reference indicia indicate like elements and in which:

FIG. 1 is a block diagram showing a network management system according to the prior art;

FIG. 2 is a block diagram showing a network management system according to an 25 embodiment of the present invention;

FIG. 3 is a block diagram showing a more detailed representation of the network management system according to an embodiment of the present invention;

FIGs. 4a-b are exemplary tags for use in a customized HTML template used in the system of FIG. 3;

FIG. 5 is a flow chart describing the operation of the services application module of 30

the system of FIG. 3; and

FIG. 6 is a flow chart describing the operation of an exemplary plug-in module in the system of FIG. 3.

5      **Detailed Description of Preferred Embodiments of the Invention**

Referring now to FIG. 1, a known system for controlling a network is shown. The system includes a plurality of network hubs HUB1, 2,...N, each hub connected to multiple network devices, such as computers, through any of a number of ports (not shown) contained in each hub. According to known network management techniques, a network administrator, 10 through a software process 10 at a designated terminal, accesses a specified management platform 12 to service the network. The management platform 12 typically dictates the type and format of information which can be retrieved from the network, and typically accesses only a single network device at a time.

Referring now to FIG. 2, a system for controlling a network according to an 15 embodiment of the present invention is shown. The system includes a similar plurality of network hubs HUB1, 2,...N, each hub connected to multiple network devices through any of a number of ports (not shown) contained in each hub. According to the system of the present invention, a web server 14 is associated with at least one of the plurality of network hubs. The web server 14 can be implemented by any standard World-Wide Web server, 20 which can support common gateway interface (CGI) 1.1. The network administrator, through software process 10 at any of the network device terminals, is able to access any combination of network devices to retrieve information in any of a wide variety of formats, regardless of the platforms of the network devices.

Referring now to FIG. 3, a more detailed block diagram of the network control 25 system of FIG. 2 is shown. As shown in FIG. 3, a network control device 30, which preferably is a network device having a web browser capability, is connected through a network hub to a computer 32 having a web server 14, a common gateway interface (CGI) module 36, and a Server Application module 38 which preferably operates according to SNMP (Simple Network Management Program), as is known in the art. The network control 30 device 30 can be a web browser of any suitable platform, including Apple, Unix or PC

platforms. The network control device 30 preferably supports hypertext markup language (HTML) 2.0, and preferably includes a Java interpreter. Any network device having a web browser capability, regardless of location, can be used as a network control device. The use of SNMP affords numerous advantages, including the capability of communicating with 5 multiple network devices to retrieve network data in real time. The SA module 38 is connected through a network hub to each of the network devices ND1, ND2,...NDi in the network. The computer 32 can further include a map manager 40 and a trap manager 42 associated with the SA module 38.

In operation, a network administrator inputs a request for network data (a navigation 10 or control request) into the network control device 30. The request is preferably in the form of a hypertext markup language (HTML) document, and is transmitted from the network control device 30 to the web server 14 of computer 32. The request can include one or more requests for, for example, a description of one or more network devices, the operational state of one or more of the network devices, any of a variety of network traffic statistics, or any 15 other relevant network information desired by the systems administrator to monitor, troubleshoot, and otherwise manage the network. The HTML document may be of a specified format, but is preferably a customized HTML document generated by the systems administrator to form an HTML template, and includes any of a number of tags, as will be explained below. The HTML template is interpreted by SA module 38 to determine the 20 information to be retrieved from the network. By using the HTML format, a graphical user interface compatible for use with the World-Wide Web can be implemented which allows a systems administrator to efficiently monitor an entire large communications network.

When the HTML template containing the request is received by the web server 14 of 25 computer 32, the CGI module 36 functions as a "traffic controller", and operates only when a request is received by the server 14 or information is retrieved by SA module 38. That is, the CGI module 36 receives the HTML template containing the request, initiates communication with the SA module 38, forwards the request to the SA module 38, and waits a predetermined time out duration for a response from SA module 38. It will be appreciated that the selective operational state of the CGI module 36 improves the efficiency of the 30 system, since CGI module 36 operates only when needed while the SA module 38 can communicate with all necessary devices to retrieve the desired information. The SA module

38 parses the HTML template containing the request to determine the nature of the request, and communicates with the appropriate physical network devices in real time to retrieve the desired information. The SA module 38 formats the retrieved information as an HTML document, and sends the retrieved information in the HTML format as a series of codewords 5 to CGI module 36 according to the following exemplary protocol. When a number of buffered data codewords are received from SA module 38 reaches or exceeds an acknowledgement increment value, CGI module 36 sends an acknowledgement increment counter. The CGI module 36 then sends all buffered data to web server 14 and sends an acknowledgement to SA module 38. The process continues until all retrieved information formatted by the SA 10 module 38 has been sent to web server 14.

As mentioned previously, computer 32 can also include a trap manager 40 and network map manager 42. As will be appreciated by those skilled in the art, when most network devices encounter an abnormal condition, they can be programmed to send a 15 notification, called a "trap", to a network management station on the network so that the administrator can respond to the abnormal condition. The system collects the traps as they are sent by network devices that experience abnormal conditions. Rather than display this information immediately, the trap manager 42 records the information in a database (not shown). This database resides on a hard disk drive 34 associated with the server 14 so the trap information is recorded permanently.

20 The trap manager 42 also allows the user to view the recorded data through web browser 30. For example, the user of control device 30 can select the trap manager 42 by clicking on a "Trap Manager" button contained in the system's "home page". Clicking on the "Trap Manager" button causes the trap manager 42 to generate a special web page by performing the operations described below.

25 The trap manager 42 examines the database created by the system on the hard disk, extracts the information stored in that database concerning traps which have occurred recently from the database, and automatically generates an HTML page which can be read by any standard browser. This page is then sent to the web server 14. The web server 14 then transmits the page to the browser 30, which draws the page. The resulting image in the 30 browser 24 shows a summary of all recent traps that have occurred in network devices on

the network. According to this exemplary embodiment, all of these operations occur automatically when a user clicks on the "Trap Manger" button on the browser 14.

In addition to the displaying of traps, the trap manager 42 allows a user to be notified of new traps received and stored by the system in the trap manager database. This 5 notification system will now be described.

When a user is operating a browser 30 and is browsing HTML pages (web pages) that are generated by the system, the system automatically downloads a Java applet to the browser 30. This Java applet is referred to hereinafter as a "notification applet". The notification applet communicates with the trap manager 42 periodically to determine if events have 10 occurred that should be automatically displayed.

When a trap is generated by a network device, it is stored in the trap manager database as described above. When this occurs, the trap manager checks to see if any notification applets are running on any browsers that are browsing HTML pages created by the system. If the system determines that any notification applets are active, it sends a 15 message to each active notification applet informing it that a trap has occurred. The notification applet receives this message and automatically displays an indication at the browser 30 which clearly indicates to the user that a trap has occurred.

The network map manager 40 is a part of the system that creates, stores, maintains, and displays information about devices present on the network. These functions will now, 20 be described.

When the user requests that the system discover all devices on the user's network, the network map manager 40 begins a process termed herein as "autodiscovery". During autodiscovery, the system polls all possible devices on the network to determine their device address and manufacturer. For each network device, the information about the device is 25 recorded at the server 14 in a network map database which resides on the hard disk 34 of the server 14. The information can be, for example, the device address and manufacturer of the device.

More particularly, to display the network map, the network map manager 40 extracts from the network map database the stored information about each device. This information 30 is converted to a graphical format which includes, for each device found on the network, a

graphic that contains a simple picture of the device, the name of the device manufacturer, and the unique internet protocol (IP) address of the device. This information is generated in HTML and the resulting web page is then sent to the web server 14. The web server 14 then transmits the web page to the browser 30. The browser 30 then displays the web page so that the user sees the graphics for all of network devices on the network.

5 After the display appears at the web browser 14, the user may click on each of the graphical images that represent each device and the display will immediately switch to a device page (see below) for that device. In this way the network map manager 40 displays a "central navigation system" for the system.

10 The entire process (including the autodiscovery process described above) occurs automatically when the user clicks on the "autodiscovery" button in the browser 30. The network map display function can be repeated by the user at a later time and certain additional information may be displayed on the screen according to an autoconfigurator function as will now be described.

15 The autoconfigurator function runs after the user first sees the graphical HTML page created by the process described above. The autodiscovery process does not gather information about the type of each network device found nor can it determine which menu options should be created for each device page (see below). The autoconfigurator function performs these operations.

20 To collect and display this information, the autoconfigurator function communicates with each device that was saved into the Network Map Database during the autodiscovery phase, and gathers information from each device (e.g., the device type and capabilities). This information is added to the network map database that was originally created in during autodiscovery (described above). Once this information is stored in the network map 25 database, it can be displayed to the user the next time the user causes the system to display the network map manager HTML pages (as described above). When the user activates the network map display to display all devices in the network, the device type information collected by the autoconfigurator function is merged with the information collected during autodiscovery, and the displayed to the user by creating HTML documents which can be 30 viewed by any web browser 30.

Another operation that can be performed by the autoconfigurator function is determining what type of device is being monitored and what network management functions are supported by the device. In order to determine what functions are supported, the network map manager 40 performs an autoconfigurator function which interrogates each 5 device to determine what type of information and features the device provides. Based on this information, network map management 40 consults a rule database contained in the system to determine, based on the information that the device supports, what capabilities (menu choices) the user should be offered when that device and its device page are displayed by the map manager 40.

10 The device page displays the information gathered from the autoconfigurator function. The device page contains a picture of the device that has been selected. Below the device picture are buttons and menu choices which allow the user to perform various network management functions on the device. The creation of the menu choices on the device page is accomplished by the SA module 38 as will now be described.

15 The information created by the autoconfigurator function is extracted from the network map database and an HTML template is created that can be read by SA module 38. The autoconfigurator information is formatted to allow the SA module 38 to interpret the information and generate an HTML page (web page) which contains the appropriate menu choices and buttons for the device. This HTML page can be read by any standard browser 20 30. This web page is then sent to the server 14.

25 The web server 14 then transmits the web page to the browser 30. The browser 30 then draws the device page so that the users sees the device's graphical image followed by the menu choices and buttons that the AutoConfigurator determined were appropriate for the device. All of these operations occur automatically when the user clicks on the devices graphical drawing shown by the network map display page.

30 Referring now to FIGs. 4a-b, exemplary tags for use in a customized HTML template in the system of FIG. 3 are shown. Specifically, one or more characters, such as "{" and/or "}" are reserved to indicate, or "tag", information such as instructions or descriptions of the type of information desired. In the exemplary tag shown in FIG. 4a, the information field "TAG" shown within the braces can be any type of information desired from the network.

Second and third information fields "INSTANCEINFO.1" and "INSTANCEINFO.2" are used in this example to indicate specific ports (referred to as "instances" in a Simple Network Management Program) from which the TAG information is desired. It will be appreciated that the number and use of information fields within each tag can be modified according to the network configuration and desired information. For example, the instance information may require an indication of a particular group of interconnected network hubs, a particular hub within the group, and a particular port within the hub. In FIG. 4b, the tags shown are exemplary tags which can be used for requesting information concerning a number of good frames GF and a number of bad frames BF generated by network devices. The asterisks "\*" in the information fields indicate that good frame and bad frame information is desired for all devices in the network. The tags shown in FIGs. 4a-b can be incorporated into an HTML document generated by the systems administrator to form all or part of a request. The tag information is replaced with the retrieved information by SA module 38 to transmit the retrieved data to the requesting systems administrator. The control system further includes supporting configuration files 44 and component plug-in modules 46. The configuration files 44 define the component plug-in modules 46 and also define control designations, as will be described below. The component plug-in modules 46 implement the features or commands of the control system. Conventional world-wide web browsers can be enhanced by plug-in modules. According to the present invention, the capabilities of the web server 14 are enhanced by the component plug-in modules 46. By installing the component plug-in modules at the server end of a world-wide web application, the enhanced capabilities become available to all browsers, as opposed to conventional systems in which only certain browsers are supplied with enhanced capabilities by plug-in modules.

The CGI module 36 enhances the capabilities of the web server 14 by translating a first type (e.g., CGI 1.1) of communications and instructions received from the browser 30 into a second type (e.g., SNMP) of communications and instructions usable by other applications (e.g., the SA module 38 and network devices ND<sub>1</sub>..ND<sub>l</sub>) which are not compatible with the first type of communications and instructions.

The functions performed by the SA module 38 will now be described. The SA module 38 extracts from the supporting configuration files 44 the component plug-in modules

46 to be managed, launches the plug-in modules 46, provides a means for allowing the plug-in modules 46 to report their operational state and processing status, and receives the operational states and processing status. The SA module 38 also extracts from the supporting configuration files 44 the control designations in a manner to be described below, provides 5 a means to communicate between the CGI module 36 and the network devices, and terminates the component plug-in modules 46 upon receipt of a termination request from the browser 30.

Referring now to FIG. 5, a flow chart describing the operation of the SA module 38 is shown. In step 100, the SA module 38 extracts the navigation and control designations 10 from the supporting configuration files 44. The designations consist of, for example, an HTML form name, an HTML form parameter designating a shared library (i.e., a memory) in which a plug-in module executable code is stored, and an HTML form parameter designating a routine within the designated shared library to be executed upon receipt of a navigation or control request by the SA module 38. In step 102, the SA module 38 15 establishes a network communications socket to monitor for navigation and control requests. In step 104, the SA module 38 receives a navigation or control request from CGI module 36, and in step 106 the SA module 38 extracts the HTML form name(s) and parameter(s) from the request. In step 108, the SA module 38 matches the extracted form name(s) and parameter(s) with the appropriate navigation and control designations extracted from the 20 supporting configuration files 44. In step 110, the SA module 38 dispatches the appropriate navigation or control request(s) to the appropriate component plug-in module(s) defined by the assigned routine(s) within the designated shared library.

An exemplary plug-in module and its operation will now be described. In this 25 example, the plug-in module adds the following features to the known HTML feature set: the ability to pass Java applet parameters; the ability to collect data in real time; the ability to retrieve data from a database; and the ability to accommodate multiple protocols. As will be appreciated by those of ordinary skill in the art, Java is a computer language frequently used in world-wide web applications. The plug-in module operates according to customized 30 HTML templates as described above with respect to FIGs. 4a-b. In this example, the information field TAG can consist of a particular shared library in the memory, a particular

routine, and a particular action. The shared library and routine can be omitted by assigning default values within the supporting configuration files 44. The plug-in module preferably performs the following functions: extracting from an HTML template a desired data collection request; dispatching the request to the indicated process; formatting the returned data; merging the returned data into the original HTML template; and forwarding the completed HTML template to the common gateway interface of the server 14 via the CGI module 36.

5 Referring now to FIG. 6, a flow chart describing the operation of the exemplary plug-in module 46 is described. In step 200, the plug-in module 46 receives a request dispatched from the SA module 38. The request includes a value identifying the HTML template to be serviced. In step 202, the plug-in module 46 extracts the identified HTML template and examines the instruction tag(s) included in the template. In step 204, the plug-in module 46 collects the requested data as indicated by the instruction tag(s). In step 206, the plug-in module 46 merges the collected data into the HTML template to create a completed HTML template, and in step 208, the plug-in module 46 forwards the completed HTML template to the server 14 via CGI module 36.

10 Thus, it will be appreciated from the foregoing description that the present invention enables a network administrator to conveniently monitor a large network which can include devices of different platforms, using a standard web browser.

15 20 While the foregoing description includes many details and specificities, it is to be understood that these are merely illustrative and are not to be construed as limitations of the invention. Those of ordinary skill in the art will recognize that many modifications can be made to the disclosed examples which do not depart from the spirit and scope of the invention, as defined by the following claims and their legal equivalents.

**WHAT IS CLAIMED IS:**

1. A system for controlling each of a plurality of network devices, comprising:
  - a first control device having a network browser for communicating with a common gateway interface of a world-wide web server; and
  - 5 a second control device for communicating with each of the plurality of network devices,
    - the common gateway interface communicating with the second control device in response to a communication received from the first control device, the second control device retrieving information requested in the communication from selected ones of the plurality of
    - 10 network devices and forwarding the retrieved information to the first control device through the common gateway interface.
2. The system of claim 1, wherein the communication is an HTML document generated by the second control device.
- 15 3. The system of claim 2, wherein the HTML document is an HTML template customized by a system user to cause the retrieved information to be displayed to the user in a desired format.
- 20 4. The system of claim 3, wherein the system user customizes the HTML template by generating one or more tags indicating a type of information desired to be retrieved from the network and one or more specific instances of the type of information desired.
- 25 5. The system of claim 1, wherein a display format for displaying the retrieved information is determined at the first control device.
6. The system of claim 1, wherein the communication identifies a network device according to its network hub and network port.
- 30 7. The system of claim 1, wherein the plurality of network devices operate according

to a plurality of operating platforms.

8. The system of claim 1, further comprising a trap manager for collecting network traps generated by the network devices.

5

9. The system of claim 1, further comprising a map manager for managing communications between each of a plurality of first control devices and the plurality of network devices.

10

10. The system of claim 9, wherein the map manager manages communications using the Internet Protocol address of each of the plurality of first control devices.

15

11. The system of claim 1, wherein a connection between the server and the first control device is disconnected after a threshold period of time if no communication is received from the first control device.

20

12. A method for retrieving information from one or more of a plurality of network devices, comprising the steps of:

25

sending a request for information from the one or more network devices from a network browser to a web server;

initiating communication between a search device and the web server in response to the request received by the network server;

forwarding the request from the web server to the search device;

30

monitoring, at the search device, the one or more network devices to retrieve the information requested in the request; and

35 forwarding the retrieved information from the search device to the network browser via the web server.

13. The method of claim 12, wherein the network server is a world-wide web server.

14. The method of claim 13, wherein the request is received at a common gateway interface of the world-wide web server.
15. The method of claim 12, wherein the request is in the form of an HTML document.
16. The method of claim 12, wherein the step of monitoring is performed by sending, in response to the request, instructions via the search device to the one or more network devices.
17. The method of claim 16, wherein the instruction identifies each of the one or more network devices by network hub and port of the network hub.
18. The method of claim 12, wherein the search device is an SNMP device.
19. The method of claim 12, further comprising the step of monitoring multiple unique sets of one or more network devices to retrieve information requested in multiple requests from multiple network browsers.
20. The method of claim 20, wherein the step of monitoring multiple unique sets is performed using an internet protocol address of each network browser.
21. The method of claim 12, further comprising the step of automatically saving retrieved information in a memory file and disconnecting the network browser from the network server.

1 / 3

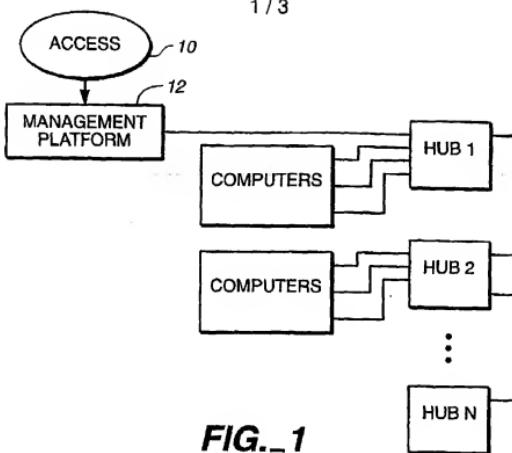


FIG. 1

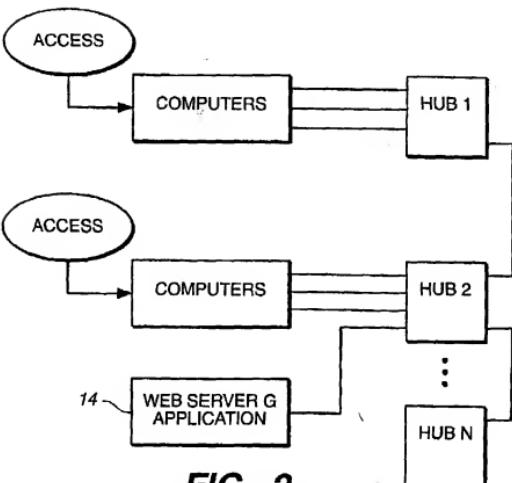
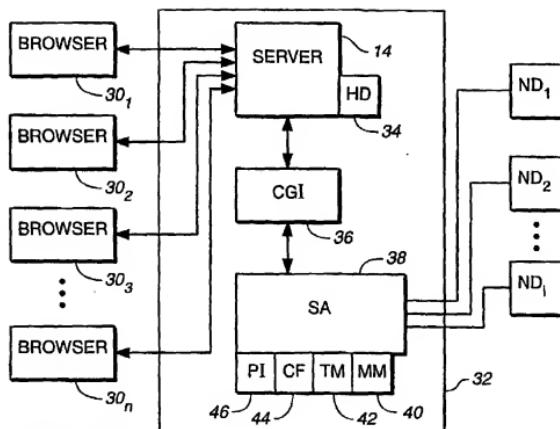


FIG. 2

SUBSTITUTE SHEET (RULE 26)

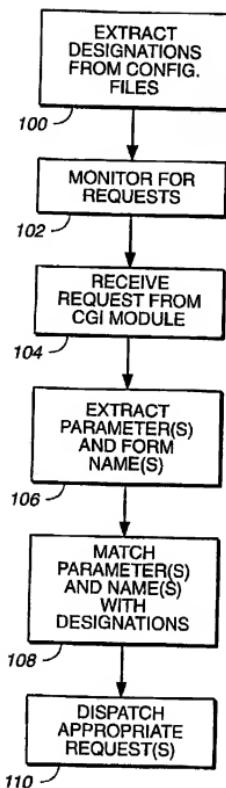
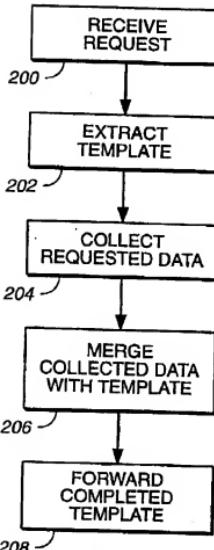
**FIG. 3**

{ TAG : INSTANCE INFO1: INSTANCE INFO2 }

**FIG. 4A**

{ (BF) : \* : \* }    ( (BF) : \* : \* )

**FIG. 4B**

**FIG.\_5****FIG.\_6**